To appear in the *International Journal of Remote Sensing*
Vol. 00, No. 00, Month 20XX, 1–20

A Supervised Hierarchical Segmentation of Remote Sensing Images using a Committee of Multi-scale Convolutional Neural Networks

Essa Basaeed[a] *, Harish Bhaskar[a], Paul Hill[b], Mohammed Al-Mualla[a], and David Bull[b]

[a] *Visual Signal Analysis and Processing (VSAP) Research Center, Khalifa University, P.O.Box 127788, Abu Dhabi, U.A.E.*; [b] *Department of Electrical and Electronic Engineering, The University of Bristol, Bristol BS8 1UB, U.K.*

(*Received 00 Month 20XX; accepted 00 Month 20XX*)

This paper presents a supervised hierarchical remote sensing image segmentation technique using a committee of multi-scale convolutional neural networks. With existing techniques, segmentation is achieved through fine-tuning a set of predefined feature detectors. However, such a solution is not robust since the introduction of new sensors or applications would require novel features and techniques to be developed. Conversely, the proposed method achieves segmentation through a set of learned feature detectors. In order to learn feature detectors, the proposed method exploits a committee of convolutional neural networks that perform multi-scale analysis on each band in order to derive individual probability maps on region boundaries. Probability maps are then inter-fused in order to produce a fused probability map. Further, the fused map is intra-fused using a morphological scheme into a hierarchical segmentation map. The proposed method is quantitatively compared to baseline techniques on a publicly-available dataset. Results, presented in this paper, highlight the improved accuracy of the proposed method.

Keywords: image segmentation; artificial neural networks; multispectral imaging; remote sensing)

1. Introduction

Image segmentation is one of the fundamental image processing techniques that facilitates autonomous image understanding in a number of application areas such as medical imaging, video surveillance, and remote sensing. Remote sensing image segmentation has evolved as a distinct research field. This evolution is driven by the complexities and distinctiveness of remote sensing data in terms of spatial, radiometric, and spectral characteristics that are uncommon to other application areas.

Despite the existence of an elaborate literature in image segmentation in general, the field of remote sensing image segmentation continues to receive growing research interest. This persistent interest stems from the continuous advancements in sensor technologies in terms of spatial, spectral, and radiometric resolutions (Li et al. 2008). Such advancements enable new applications which consequently require novel segmentation techniques in order to cope with the unprecedented re-

*Corresponding author. Email: ebasaeed@kustar.ac.ae

quirements. Furthermore, remote sensing applications often require images to be analysed at different scales (e.g., forest-level and tree-level). However, there are three limitations that affect the general applicability of segmentation techniques: 1) the explicit set of homogeneity measures in the feature space that encode the characteristics of the object of interest, 2) the selection of optimal parameters for the segmentation process, and 3) single-scale analysis of imagery data. However, most segmentation techniques address few of but not all of these limitations. For example, the first limitation is addressed using a statistical analysis or machine learning techniques such as iterative self-organizing data analysis (Jensen 2004), self-organizing maps (Visa, Valkealahti, and Simula 1991), and pulse-coupled neural networks (Li, Ma, and Wen 2007). The second limitation is tackled using the simplistic trial-and-error approach (Trias-Sanz, Stamon, and Louchet 2008; Kim et al. 2011; Myint et al. 2011) or an optimization algorithm (Novack et al. 2011). Finally, the third limitation is addressed producing a hierarchical segmentation via processing the image at different scales (Guigues, Cocquerez, and Men 2006), at different values of a homogeneity criterion (Salembier and Garrido 2000), at different values of a dissimilarity criterion (Tilton et al. 2012), or following the Markov random field (MRF) model (D'Elia, Poggi, and Scarpa 2003). As such, the scale that fits a particular application can be set later and the segmentation remains scale uncommitted (Guigues, Cocquerez, and Men 2006; Tarabalka et al. 2012; Hu et al. 2013). Therefore, there is a need for a segmentation method that addresses all limitations for it to adapt to different requirements with minimum user intervention.

In this paper, a novel structure of a committee of Multi-Scale Convolutional Neural Networks (MSCNNs) is proposed for remote sensing image segmentation. The proposed method can seamlessly integrate spectral and spatial information while providing a hierarchical segmentation. In addition, it requires minimum user intervention. The structure is novel in essence that it introduces multi-scale image analysis within the framework of image segmentation that is based on Convolutional Neural Networks (CNNs) with no increase in computational complexity of the CNN compared to the single-scale counterpart and no redesigning burden in terms of the CNN architecture. Moreover, the work described in this paper is one among the first remote sensing image segmentation techniques that incorporate CNNs.

The paper is organized as follows. In the following section, relevant background information as well as related work in the literature are highlighted. In section 3, the problem of image segmentation is mathematically formulated. It also describes the proposed method in detail. Section 4 presents a quantitative comparison between the proposed method and other baseline techniques in addition to the effect of boosting on the performance of the proposed method. The last section concludes with highlights on future directions.

2. Background and Related Work

At the core of the proposed method is the use of CNNs. A CNN is a machine learning technique that consists of a hierarchy of layers of three different types, namely, convolutional, pooling, and fully-connected layers. Convolutional layers are the main layers in a CNN architecture. It consists of a set of filters that act as feature detectors. Pooling layers perform sub-sampling of data and hence, force subsequent layers in the network to focus on features at a coarser scale. In addition,

pooling layers add a spatial-invariance property (Scherer, Mller, and Behnke 2010). While sub-sampling is crucial, the importance of the spatial-invariance property is application-dependent. Finally, fully-connected layers are usually the last few layers in a CNN and they are the classification layers. The input of a CNN is a raw image and the output is a classification.

CNN has been used in order to improve classification results in a number of applications (Ciresan et al. 2012; Chen et al. 2014b; Collobert and Weston 2008; Abdel-Hamid et al. 2012). In what follows, a few representative applications of CNN in image classification and segmentation across different fields are highlighted. One application is to use CNN in order to classify whole images into different object classes as in (Krizhevsky, Sutskever, and Hinton 2012) and localize objects as in (Schulz and Behnke 2012) and (Long, Shelhamer, and Darrell 2015). In medical imaging, CNN is used as a pixel classifier for the purpose of detecting mitosis in breast cancer histology images as demonstrated in (Ciresan et al. 2013). In addition, they are used in order to classify gray matter, white matter, and cerebrospinal fluid in infant brain tissue images as in (Zhang et al. 2015). In field of remote sensing, CNN is proposed for the classification of roads and buildings in aerial images in (Mnih 2013), the classification of different land covers in hyper-spectral remote sensing data in (Chen et al. 2014b; Yue et al. 2015), and the detection of vehicles in high-resolution satellite images in (Chen et al. 2014a). In (Penatti, Nogueira, and dos Santos 2015), CNNs trained to classify generic images are used in order to distinguish different land uses in remote sensing images. In (Zhao et al. 2015), for the purpose of classifying hyper-spectral images, CNNs are used to extract features at different scales from a pyramid image. Outputs of the CNNs are up-sampled in order to match the size of the original image. Then, these outputs (features) along with principal component analysis features are classified using the logistic regression classifier.

The work presented in this paper is motivated by the work of Ciresan et al. (2012). In (Ciresan et al. 2012), four CNNs are used in order to classify the pixels in a single-band electron microscopy image into membrane and non-membrane pixels. The input image is sliced into small patches using a sliding window. Each CNN is configured with a different architecture. The reason behind changing the architecture is to force different CNNs to extract different features and hence to produce complementary outputs. Each CNN will produce a confidence map in which each pixel represents the probability of being a membrane pixel. The final classification map is obtained by applying a threshold on the average of all CNN outputs.

3. Proposed Method

3.1. *Problem Formulation*

As illustrated in Figure 1, let $\mathrm{I} = \{p(x, y) : 1 \leq x \leq X, 1 \leq y \leq Y\}$ where $p(x, y)$ is the value of a pixel at the Cartesian position (x, y) in a single-band image, I, of $X \times Y$ resolution. A segmentation can be achieved by, first, deducing a function, $\varphi : p \rightarrow [0, 1]$ where $\varphi(p(x, y))$ is closer to 1 if (x, y) is a boundary pixel and closer to 0 otherwise. Let $\boldsymbol{S} = \{s_1, s_2, \cdots, s_M\}$ represent a segmentation over an image I such that $\cup_m s_m = \mathrm{I}$ and $\forall(m, n), m \neq n \Rightarrow s_m \cap s_n = \phi$ where $\cup$ and $\cap$ are the union and intersection operators, respectively. In order to produce a hierarchical segmentation, a partitioning function, A_λ, is applied as in $A_\lambda : \varphi \rightarrow \boldsymbol{S}$ where λ is the scale parameter. λ is called a scale parameter if and only if $\forall(\lambda_1, \lambda_2) \in$

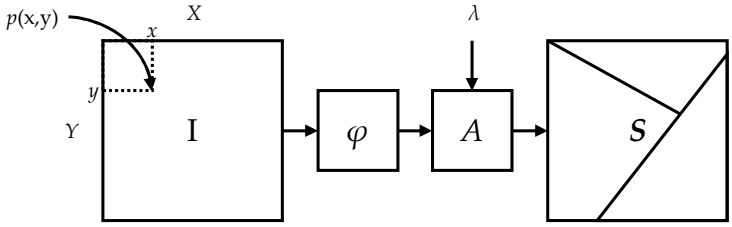


Figure 1. A flow-diagram of different steps in segmentation from the input image towards producing a segmentation map.

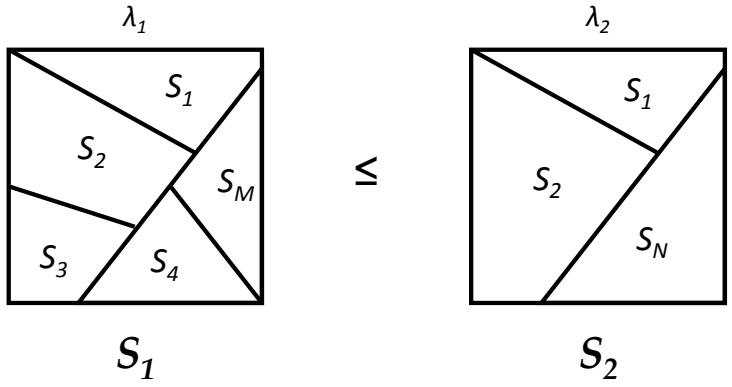


Figure 2. An illustrative example of two different segmentation maps produced from a hierarchical segmentation as a result of changing the scale parameter; the segmentation map on the left is considered finer than the segmentation map on the right.

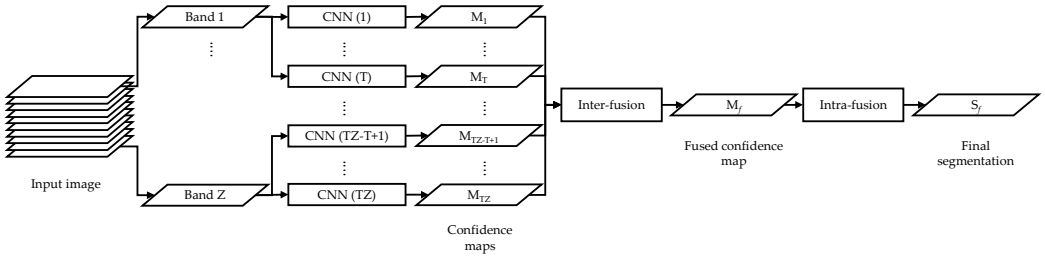


Figure 3. The conceptual work flow of the proposed method.

$R^{+^2}, \lambda_1 \leq \lambda_2 \Rightarrow \boldsymbol{S}_1 \leq \boldsymbol{S}_2$, where $\boldsymbol{S}_1 \leq \boldsymbol{S}_2$ indicates that $\boldsymbol{S}_1$ is finer than $\boldsymbol{S}_2$ and defined as $\forall s_m \in \boldsymbol{S}_1, \exists s_n \in \boldsymbol{S}_2 : s_m \subseteq s_n$ (refer to Figure 2).

3.2. *Method Formulation*

The conceptual work flow of the proposed method is shown in Figure 3. In order to extend the single-band formulation (in Section 3.1) to multi-spectral bands, let $\mathrm{B} = \{\mathrm{I}_z \mid 1 \leq z \leq Z\}$ where B is a Z-band image and I_z represents a single band in the image. Also, a patch, $\boldsymbol{p}(x, y)$, is defined as $\boldsymbol{p}(x, y) = \{p(x - \lfloor w/2 \rfloor, y - \lfloor w/2 \rfloor), \ldots, p(x + \lfloor w/2 \rfloor, y + \lfloor w/2 \rfloor)\}$ which denotes the $w \times w$ window surrounding the pixel at the Cartesian coordinates (x, y) in a band. Also, w should be an odd number for the window to have a centre pixel. This condition is necessary as the proposed method classifies the centre pixel. First, each band, I_z, is divided into a set of overlapping patches of different window sizes similar to a Gaussian pyramid

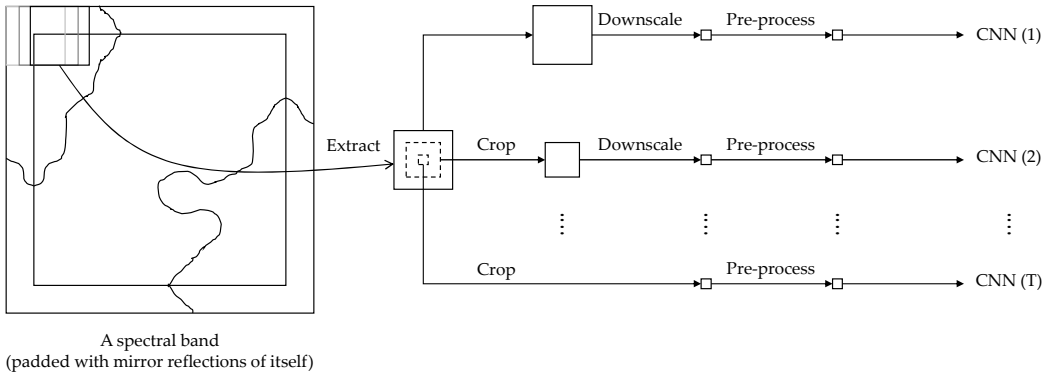
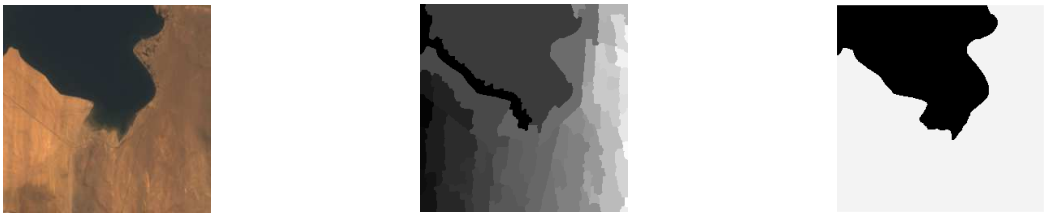


Figure 4. The process of extracting multi-scale input patches from a spectral band.



(a) (b) (c)

Figure 5. The effect of introducing multi-scale analysis in the segmentation framework; (a) an input image, (b) single-scale CNNs, and (c) MSCNNs.

as in:

$$w_t = \begin{cases} w_b, & \text{if } t = T \\ 2^{T-t} w_b - 1, & \text{otherwise} \end{cases} \quad (1)$$

where T denotes the total number of CNNs per band and w_b is the base patch size. In order to extract patches around the boundaries of the band, the size of the band is increased by mirroring pixel values across its boundary, as illustrated in Figure 4. Then, large patches are resized to the base patch size ($w_b \times w_b$). Figure 4 demonstrates the process of extracting multi-scale patches from a spectral band. Since all patches are resized to the same size, the same CNN structure can be used across different scales. The importance of introducing multi-scale analysis within the segmentation framework is demonstrated in Figure 5. In the centre of the input image (Figure 5(a)), there is a smooth change from the land-class to the water-class. While the single-scale CNNs managed to detect several boundaries, the smooth change is difficult to detect. Incorporating multi-scale analysis, the land-water boundary is easier to detect as a result of larger context.

The training set of a network is the set of patches $\{\boldsymbol{p}(1,1), \boldsymbol{p}(1,2), \ldots, \boldsymbol{p}(X,Y)\}$ extracted from a particular band at a particular scale along with their corresponding reference labels for the centre pixel being a boundary or a non-boundary pixel, $\{l(1,1), l(1,2), \ldots, l(X,Y)\}$. During testing, patches are extracted in a similar approach as with training and input into the committee of CNNs. Each network will produce a confidence map $\mathrm{M}_c = \{\varphi_c(\boldsymbol{p}(x,y)) : \forall x \in X, y \in Y\}$, as illustrated in Figure 3. All confidence maps are then inter-fused to produce a single confidence

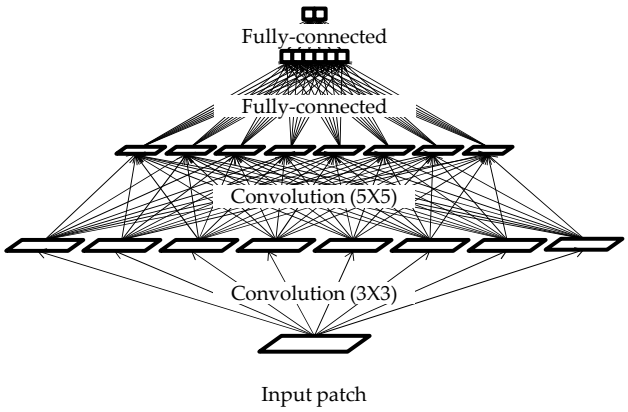


Figure 6. The architecture of the CNN in the proposed method.

(*a*) (*b*) (*c*)

Figure 7. A boundary (positive) patch where the centre pixel is a boundary pixel – enlarged for visualization purposes – extracted at different scales; (*a*) 59×59 downscaled to 15×15 (coarse), (*b*) 29×29 downscaled to 15×15, (*c*) 15×15 (fine).

map (refer to Figure 3) defined as:

$$\mathrm{M}_f = \sum_{c=1}^{TZ} \alpha_c \mathrm{M}_c \tag{2}$$

where T denotes the total number of CNNs per band, Z is the total number of bands in the multi-spectral image, and α_c is a user-set fusion parameter. Further, the fused confidence image is intra-fused using H-minima and the watershed transform. Intra-fusion aims to thin boundaries, connect them into closed contours, and produce a hierarchical segmentation map. First, H-minima transform is applied on M_f as in (Jung and Kim 2010):

$$H_\lambda(\mathrm{M}_f) = R^{\varepsilon}_{\mathrm{M}_f}(\mathrm{M}_f + \lambda) \tag{3}$$

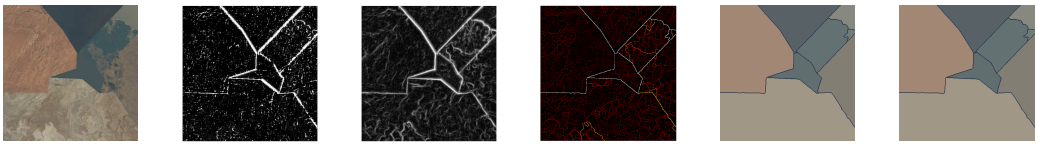
where λ is a user-set depth parameter and acts as a scale parameter, R is the reconstruction operator, and ε is the erosion operator. Further, the watershed transform is applied in order to produce the final segmentation, $\boldsymbol{S}_f$, where watershed lines would represent region boundaries.

3.3. *Method Implementation*

The first step is to set the CNN architecture. The proposed CNN architecture can be visualized in Figure 6. In details, the CNN takes a patch of 15×15 pixels as input, processes it through a convolution layer that consists of 8 filters each of which is a 3×3 filters applied at a 2×2 stride, passes the results of the first

(a) (b) (c)

Figure 8. A non-boundary (negative) patch where the centre pixel is not a boundary pixel – enlarged for visualization purposes – extracted at different scales; (a) 59×59 downscaled to 15×15 (coarse), (b) 29×29 downscaled to 15×15, (c) 15×15 (fine).



(a) (b) (c) (d) (e) (f)

Figure 9. A walk through example of the inter-fusion and intra-fusion steps in the proposed method; (a) the input image, (b) a confidence image produced by a single CNN, (c) the inter-fused confidence image, (d) contour disappearance map produced by intra-fusion with different depth (scale) values, (e) the segmentation map at the optimal scale parameter, and (f) is the final segmentation map after merging.

convolution layer to another convolution layer of 8 maps each of which is of 5×5 applied at a 2×2 stride, forwards the results to a fully connected layer of 6 hidden units, and then forwards the results to another fully connected layer of two nodes where each of which represents a class being boundary or non-boundary patch. For the activation functions, a rectified linear function is used for convolution layers whereas a softmax function is used for the fully connected layers. Following the convention in the literature as in (Ciresan, Meier, and Schmidhuber 2012; Wu and Gu 2015), the aforementioned architecture can be abbreviated as 1×15×15-8C(2)3-8C(2)5-6F-2F. As with other work in the literature, the architecture is set after a series of empirical tests through potential architectures following the recommendations in (Simard, Steinkraus, and Platt 2003). As such, the base patch size, w_b, is set to 15 which is the input of the CNN. Since all multi-scale patches are downsized to the same base size, all CNNs in the committee can share the same architecture.

CNN training starts by separating bands of the remote sensing image. Then, overlapping patches from each band in each image are extracted with three different sizes (as defined in Eq. 1): 15×15, 29×29, and 59×59. Then, the two larger patches are downsized to 15×15 using the bicubic transformation. As such, the resultant patches are of the same size (15×15) but display different contexts. Figures 7 and 8 show a boundary (positive) patch where the centre pixel is a boundary pixel and a non-boundary (negative) patch where the centre pixel is a non-boundary pixel, respectively. The figures also demonstrate the different context that patches exhibit across scales. Each network shall receive patches in a particular scale extracted from a particular band. When training, however, the number of positive (boundary) patches are far less than negative (non-boundary) patches. If trained with such an unbalanced dataset, the network will be biased towards non-boundary patches and fail to generalize. Therefore, negative samples are randomly selected with no repetition in order to match the number of positive samples. Also, in order to avoid edge pixels to be at the centre of the patch at the coarser scale, no negative sample is extracted in the close proximity of positive samples. Then, both positive and

negative samples are randomly left as is, rotated at $\pm 90^\circ/180^\circ$, or flipped across the horizontal and vertical axes. This step aims to introduce invariance to such changes in the network. Further, pixels at the same location across all patches are normalized to zero mean and unit variance. After these pre-processing steps, patches are input into the CNN for training.

After training, each of the CNNs in the committee will result in a confidence map similar to the one in Figure 9(b). The confidence maps produced by the whole committee of CNNs are inter-fused with the mean fusion scheme: $\alpha_c = 1/TZ$ in Eq. 2, where T denotes the number of CNNs per band and Z denotes the total number of bands, and would result in a fused confidence map as shown in Figure 9(c). Then, the fused confidence map is intra-fused in order to produce a hierarchical segmentation as the depth (scale) parameter is varied (refer to Figure 9(d) which shows how the contours disappear as a result of increasing the value of the depth (scale) parameter). The selection of the optimal scale, λ, is application-dependent. Here, λ is set in order to produce the closest result to the reference segmentation. The result is shown in Figure 9(e). In order to enhance results further, regions whose sizes are below a threshold (called the merging threshold) are merged with a neighbouring region with which it shares the longest boundary. The merging process is repeated until no region has a size below the threshold. The merging process is not critical and, for example, no merging is performed for the demonstrated case in Figure 9. The final segmentation is depicted in Figure 9(f).

For the particular case in this paper, there are thirty CNNs in the committee since images in the Prague texture dataset have ten bands, $Z = 10$, and each band is assigned three CNNs, $T = 3$. The reason of assigning three CNNs per band is to allow boosting in training as documented in further detail in Section 4.3. The training set consists of all boundary patches in the reference segmentation map of the 30 (75%) training images and an equal number of randomly selected non-boundary patches (a total of approximately 150,000 patches). In order to assure a proper configuration, none of the patches are extracted from the 10 (25%) testing images. This number of samples is found to be sufficient for such a number of layers as in other works in CNN documented in the literature (Chen et al. 2014a; Lecun et al. 1998; Sermanet and LeCun 2011). Results generated show that fewer number of training data (100,000 patches) did not affect the quality of the proposed method in terms of the Correct Segmentation metric (CS) but a further reduction (to 50,000 patches) degraded the segmentation quality by 10%.

4. Experiments and Results

4.1. *Data Set*

For the purpose of evaluation, the Prague texture dataset is used (Mikes, Haindl, and Scarpa 2012). The dataset comprises of a set of synthetic mosaics of textures. These textures are extracted from real remote sensing images captured using the Advanced Land Imager (ALI) (please refer to Ungar et al. 2003, for further details on ALI). The dataset consists of 40 images out of which 10 are reserved for testing. In Figures 10(a)–,(e) and Figures 11(a)–,(e), the 10 test images from the dataset are depicted. Each image is a 10-band image of 512×512 resolution. The dataset provides a test bed to evaluate different segmentation techniques. Textures in the dataset are natural but region boundaries are not.

Since boundaries in the Prague texture set are not natural, the dataset is complemented with a set of real remote sensing images. There are 9 images in total in the

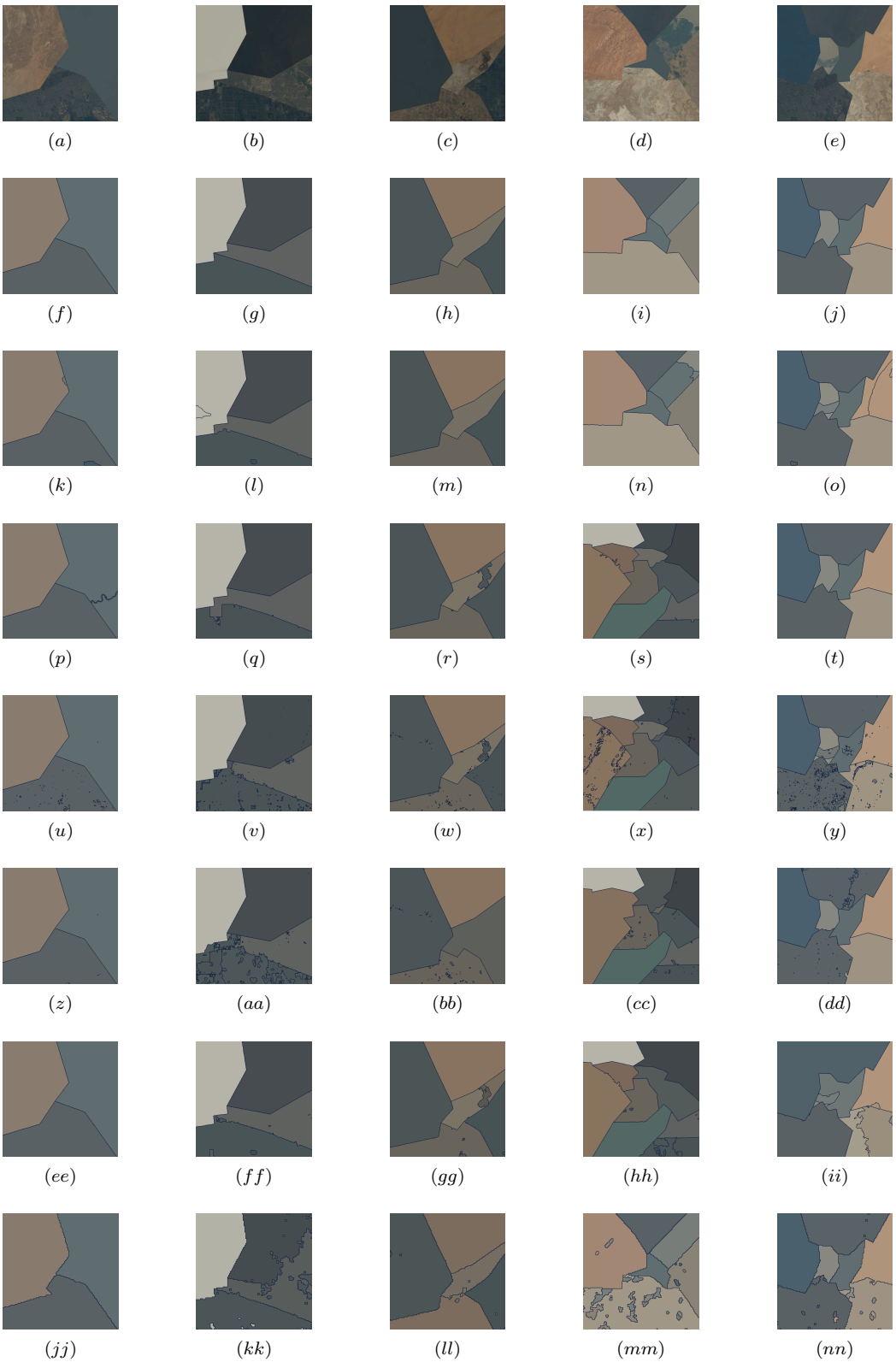


Figure 10. Segmentation results on the Prague dataset (part I); (a)–,(e) input images, (f)–,(j) reference segmentation, (k)–,(o) MSCNNs , (p)–,(t) eCognition, (u)–,(y) the Dynamic Hierarchical Classification (DHC), (z)–,(dd) Recursive Texture Fragmentation and Reconstruction (R-TFR), (ee)–,(ii) ENVI Feature Extraction (ENVI), and (jj)–,(nn) Neuralnet.

Figure 11. Segmentation results on the Prague dataset (part II); (a)–,(e) input images, (f)–,(j) reference segmentation, (k)–,(o) MSCNNs , (p)–,(t) eCognition, (u)–,(y) the Dynamic Hierarchical Classification (DHC), (z)–,(dd) Recursive Texture Fragmentation and Reconstruction (R-TFR), (ee)–,(ii) ENVI Feature Extraction (ENVI), and (jj)–,(nn) Neuralnet.

Table 1. Meta-data of images in the complementary dataset

Image label in Figure 14	Entity ID	Location
(a)	EO1A2160142008126110PP_SGS_01	Eastern region, Iceland
(b)	EO1A0120312001129111P1_PF1_01	Massachusetts, USA
(c)	EO1A1600432003100110PZ_PF1_01	Abu Dhabi, UAE
(d)	EO1A0890792001252111PP_AGS_01	Brisbane, Australia
(e)	EO1A0070692008232110K1_SGS_01	Ica region, Peru
(f)	EO1A0370232001128111PP_SGS_01	Saskatchewan, Canada

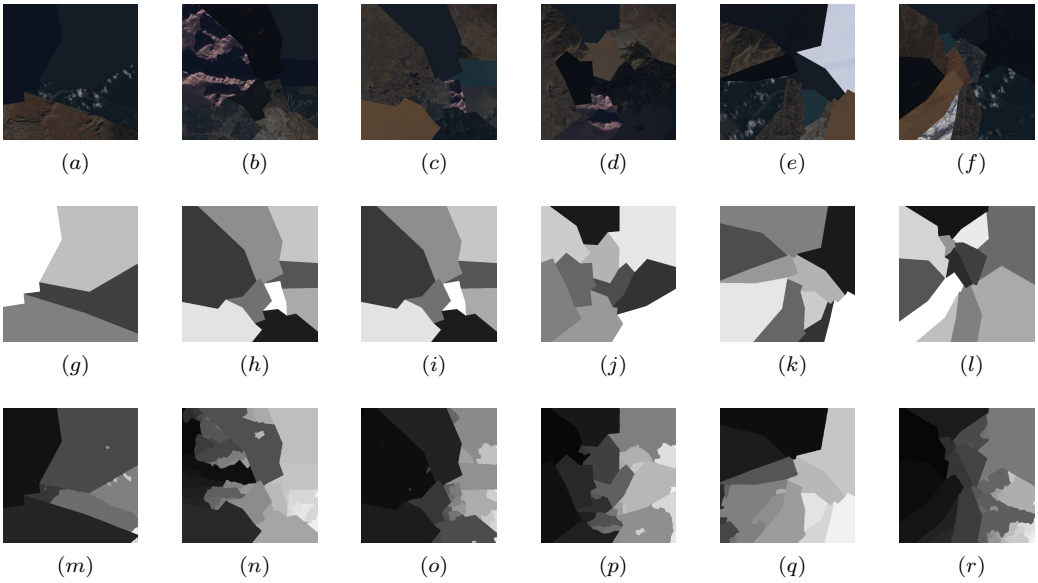


Figure 12. A list of images where the proposed method performs poorly on the cross-validated Prague texture set; (a)–,(f) input images, (g)–,(l) reference segmentation maps, (m)–,(r) MSCNNs.

complementary dataset. Six images captured using ALI whereas the other three are captured by WorldView-2 (shown in Figures 14(a)–,(f) and Figures 16(a)–,(c), respectively). These sensors vary in their spectral, radiometric, and spatial characteristics. The former is a sensor of a medium spatial resolution (30m) (Weng and Hu 2008) whereas the latter is a sensor of a high spatial resolution (0.5m). It is worth noting that WorldView-2 images that are used in this paper are limited to 3 bands due to data availability. The complementary dataset would provide an insight into the performance of the proposed method on curved region boundaries as well as its applicability on images captured using other sensors. Since reference segmentation maps are not available for the complementary dataset, the evaluation shall be merely subjective. Images in the complementary dataset are publically-available. For easy retrieval from Earth Explorer[1], entity identification numbers of images in the complementary dataset are listed in Table 1.

4.2. *Comparative Analysis*

A set of well-established metrics is used on the Prague texture set in order to compare the proposed method with baseline techniques. The selected techniques include top-rated commercial segmentation techniques (Marpu et al. 2010), namely, Definiens Developer (a.k.a, eCognition) (Baatz and Schape 2000) and

[1]http://earthexplorer.usgs.gov/

ENVI Feature Extraction Module (ENVI) (Xiaoying 2009). In addition, more recent techniques in the literature such as the Dynamic Hierarchical Classification (DHC) (Scarpa et al. 2013) and Recursive Texture Fragmentation and Reconstruction (R-TFR) segmentation techniques (Gaetano, Scarpa, and Poggi 2009) are considered. All these methods are unsupervised but they act as important benchmarks in the field of remote sensing image segmentation. Comparing against them demonstrate the significance of the proposed method. Neuralnet method (Scarpa et al. 2012) is a supervised method where it classifies the regions in order to derive the segmentation. This method requires the number of classes as *a priori* knowledge. The parameters of all methods are manually set in order to provide the best segmentation for each individual image in the test dataset. The same applies for the selection of the scale parameter, λ, in the proposed method whereas the merging threshold is set to a fixed value that is 300. In order to avoid any bias, results of the proposed method documented in Table 2 are the average of five runs. Results show that the proposed method ranks the best among all baseline techniques. Subjectively, results in Figures 10 and 11 confirm the accuracy of the proposed method among other techniques. In addition, the proposed method has produced a segmentation that is exactly the reference in at least one occasion whereas others are highly similar. It is worth mentioning that results of the proposed method reported in Figure 10 and 11 are of the one of median performance among the five runs according to correct segmentation (CS) metric. This is to assure a fair visual comparison to other techniques.

Furthermore, a 4-fold cross-validation is performed on the proposed method, since it is a supervised segmentation method, using the Prague texture set. The cross-validation test provides an insight into the performance of the proposed method on an independent set. Test results indicate that the proposed method has a consistent performance across all metrics (with a mean correct segmentation (CS) of 92.63%). However, there are few examples where the proposed method produces a segmentation that is dissimilar to the reference segmentation. These worst cases (as per the Correct Segmentation (CS) metric) are presented in Figure 12. It can be noticed that the proposed method fails on particular regions such as the cloudy region in Figures 12(a), (e), & (f) and the mountainous region in Figures 12(b)–,(d). These regions are difficult to segment as they exhibit within different land covers and shadow areas (shadows have a spectral response similar to water (Basaeed, Bhaskar, and Al-Mualla 2013)). In addition, rivers can lead to over-segmentation as observed in the top right region in Figure 12(d). This is because the proposed method is an edge-based segmentation technique and rivers would cause a strong response (boundary) of the trained CNNs.

4.3. *Method Variations*

It is anticipated that using boosting when training could improve the overall accuracy of the committee. Boosting is performed on the three CNNs that are assigned for each band (i.e., across scales of the same band). In boosting, the full set of patches and the corresponding reference values are passed to the first network during training. The first network would receive the whole training set, $\boldsymbol{P}_1$, learn the function φ_1, and produce a predicted label, $E_1(x, y)$, for each input patch, $\boldsymbol{p}(x, y)$, using the following equation:

$$E_1(x,y) = \begin{cases} 1 & \text{if } \varphi_1(\boldsymbol{p}(x,y)) \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Table 2. Quantitative evaluation results of different baseline techniques on the Prague dataset

Evaluation metric	MSCNNs	eCognition[a]	DHC[a]	Neuralnet[a]	R-TFR[a]	ENVI[a]
Correct segmentation (%)(CS↑)	**94.52**	91.91	84.60	79.85	78.45	73.49
Over-segmentation (%)(OS↓)	25.16	10.54	6.78	**3.38**	10.39	24.01
Under-segmentation (%)(US↓)	**0.00**	1.11	8.73	13.52	14.33	16.74
Missed error (%)(ME↓)	**0.00**	1.20	4.70	2.82	2.90	5.46
Noise error (%)(NE↓)	**0.00**	0.98	5.33	3.21	1.28	4.71
Omission error (%)(O↓)	0.64	**0.06**	1.69	2.74	1.22	2.33
Commission error (%)(C↓)	5.23	**0.52**	0.70	3.27	3.11	80.45
Class accuracy (%)(CA↑)	**95.80**	94.13	89.69	84.36	84.74	81.53
Recall - correct assignment (%)(CO↑)	**96.39**	95.42	92.80	90.56	89.72	86.48
Precision - object accuracy (%)(CC↑)	**99.39**	98.16	92.78	88.37	88.98	88.25
Type I error (%)(I.↓)	**3.61**	4.58	7.20	9.44	10.28	13.52
Type II error (%)(II.↓)	**0.13**	0.24	0.90	1.89	1.25	1.67
Mean class accuracy estimate (%)(EA↑)	**97.46**	96.13	92.29	88.81	88.37	85.52
Mapping score (%)(MS↑)	**96.08**	94.40	89.59	85.84	85.45	81.87
RMS proportion estimation error (%)(RM↓)	**0.89**	1.62	1.94	3.34	3.46	2.79
Comparison index (%)(CI↑)	**97.66**	96.44	92.53	89.13	88.84	86.38
Global Consistency Error (%)(GCE↓)	**1.22**	2.67	4.38	7.23	4.34	5.76
Local Consistency Error (%)(LCE↓)	**0.80**	1.16	2.67	4.89	2.55	1.98
Van Dongen metric (%)(dD↓)	**2.13**	2.91	4.61	6.81	6.20	7.58
Mirkin metric (%)(dM↓)	**0.94**	1.24	2.17	4.61	3.06	4.13
Variation of information (dVI↓)	14.88	14.75	14.51	14.51	**14.43**	14.79

Note: the best results are highlighted in bold typeface.
[a]Results documented in this table are retrieved from: http://mosaic.utia.cas.cz/
[b]Root mean square proportion estimation error.

Table 3. Quantitative evaluation results of different method variations on the Prague dataset

Evaluation metric	MSCNNs μ (σ)	Boosted MSCNNs μ (σ)	CNNs μ (σ)	Boosted CNNs μ (σ)
Correct segmentation (%)(CS↑)	**95.19** (0.92)	94.52 (1.58)	93.32 (2.32)	93.98 (1.71)
Over-segmentation (%)(OS↓)	24.70 (3.56)	25.16 (2.28)	14.06 (2.39)	**12.38** (3.18)
Under-segmentation (%)(US↓)	**0.0** (0.0)	**0.0** (0.0)	0.75 (1.68)	**0.0** (0.0)
Missed error (%)(ME↓)	0.19 (0.42)	**0.0** (0.0)	0.81 (1.80)	2.19 (2.69)
Noise error (%)(NE↓)	0.07 (0.16)	**0.0** (0.0)	0.80 (1.80)	1.91 (2.37)
Omission error (%)(O↓)	**0.63** (0.06)	0.64 (0.07)	0.98 (1.16)	0.76 (0.42)
Commission error (%)(C↓)	7.80 (6.57)	**5.23** (5.35)	17.29 (10.96)	9.64 (3.74)
Class accuracy (%)(CA↑)	**96.40** (0.52)	95.80 (0.69)	95.62 (1.68)	95.97 (1.35)
Recall - correct assignment (%)(CO↑)	**96.85** (0.44)	96.39 (0.67)	96.20 (1.50)	96.71 (0.96)
Precision - object accuracy (%)(CC↑)	**99.53** (0.20)	99.39 (0.06)	99.20 (0.68)	99.24 (0.60)
Type I error (%)(I.↓)	**3.15** (0.44)	3.61 (0.67)	3.80 (1.50)	3.29 (0.96)
Type II error (%)(II.↓)	**0.13** (0.09)	0.13 (0.0)	0.18 (0.15)	0.26 (0.28)
Mean class accuracy estimate (%)(EA↑)	**97.82** (0.30)	97.46 (0.44)	96.93 (1.37)	97.40 (0.93)
Mapping score (%)(MS↑)	**96.61** (0.47)	96.08 (0.68)	95.84 (1.61)	96.26 (1.21)
RMS proportion estimation error[b] (%)(RM↓)	**0.83** (0.15)	0.89 (0.24)	0.93 (0.35)	1.02 (0.29)
Comparison index (%)(CI↑)	**98.00** (0.28)	97.66 (0.39)	97.28 (1.15)	97.67 (0.82)
Global Consistency Error (%)(GCE↓)	**0.88** (0.24)	1.22 (0.12)	0.93 (0.65)	1.39 (0.89)
Local Consistency Error (%)(LCE↓)	**0.60** (0.09)	0.80 (0.07)	0.61 (0.34)	0.83 (0.43)
Van Dongen metric (%)(dD↓)	**1.80** (0.25)	2.13 (0.34)	2.18 (0.92)	2.07 (0.68)
Mirkin metric (%)(dM↓)	**0.85** (0.24)	0.94 (0.19)	1.30 (0.50)	1.15 (0.54)
Variation of information (dVI↓)	14.81 (0.07)	14.88 (0.08)	14.90 (0.25)	**14.75** (0.11)

Note: the best results are highlighted in bold typeface.
*μ and σ represent the mean and the standard deviation respectively.
[b]Root mean square proportion estimation error.

The predicted labels, E_1, are cross-checked with reference values, l, and then separated into two sets, namely, correctly classified $R_1 = \{\boldsymbol{p}(x,y) : \boldsymbol{p}(x,y) \in \boldsymbol{P}_1 \mid E_1(x,y) = l(x,y)\}$ and misclassified datasets $R_1^{\complement} = \{\boldsymbol{p}(x,y) : \boldsymbol{p}(x,y) \in \boldsymbol{P}_1 \mid E_1(x,y) \neq l(x,y)\}$. The next network receives only the set of patches $\boldsymbol{P}_2$ such that $\mathrm{card}(\{\boldsymbol{p}(x,y) : \boldsymbol{p}(x,y) \in \boldsymbol{P}_2 \mid \boldsymbol{p}(x,y) \in R_1\}) = \mathrm{card}(\{\boldsymbol{p}(x,y) : \boldsymbol{p}(x,y) \in \boldsymbol{P}_2 \mid \boldsymbol{p}(x,y) \in R_1^{\complement}\})$ and $\mathrm{card}(\boldsymbol{P}_2) = 2 \times \min(\mathrm{card}(R_1), \mathrm{card}(R_1^{\complement}))$ where $\mathrm{card}(S)$ is the cardinality of a set S. The output of the second network is similarly assessed and predicted labels are separated into correctly classified R_2 and incorrectly classified

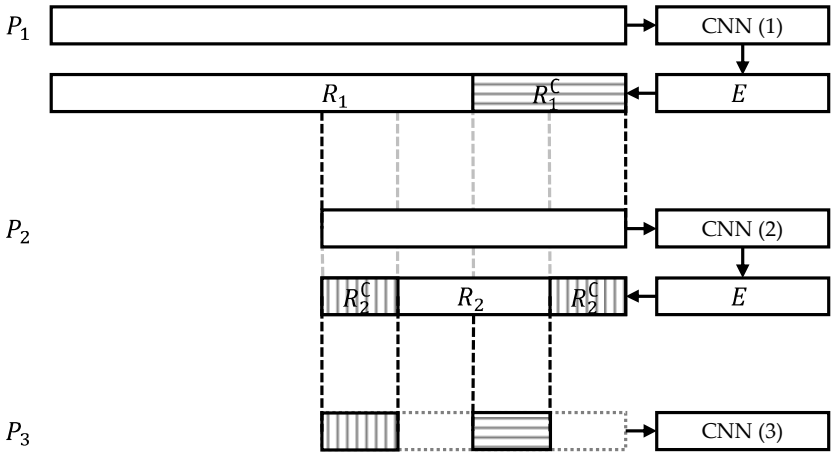


Figure 13. An illustration of the process of boosting where the training dataset gets reduced from one CNN to the next in order to allow the following CNN to focus on more difficult samples.

$R_2^{\complement}$ sets. The input of the third and final network is $\boldsymbol{P}_3 = (R_1 \cap R_2^{\complement}) \cup (R_2 \cap R_1^{\complement})$. That is to say that boosting passes difficult patches at different scales from one network to the other and the third network aims to resolve disagreement between the first and the second networks. In all cases, the raining set should be balanced where negative samples are equal to positive samples. Once trained, each network shall independently estimate the function $\varphi_c : \boldsymbol{p} \rightarrow [0, 1]$ where c indicates a particular network and $\varphi_c(\boldsymbol{p}(x, y))$ would result in a high confidence value if (x, y) is a boundary pixel and a low confidence value otherwise. The process is repeated in a similar manner across all bands.

In this set of experiments, four versions of the proposed method are tested on the Prague dataset: Multi-Scale CNNs (MSCNNs), single-scale CNNs (CNNs), and each of which with and without boosting. The results, summarized in Table 2, show that the MSCNNs (1.29) is the best among the different variations of the proposed method followed by boosted MSCNNs (2.29), boosted CNNs (2.71), and CNNs (3.52) according to the (average rank) across all metrics. However, the boosted MSCNNs is the best in terms of stability since having the lowest average standard deviation (σ) in metrics across the five runs. Boosted MSCNNs is followed by MSCNNs, boosted CNNs, and CNNs, receptively. Thus, it can be said that multi-scale analysis improves the performance whereas boosting improves the stability. In addition, boosting reduces the computational complexity in subsequent CNNs as a result of data reduction. However, boosting makes learning a sequential process for each band. In other words, the proposed method with no boosting can run in parallel using 30 CPU cores where each core will train a single CNN. On the other hand, the proposed method with boosting can run in parallel using 10 CPU cores where each core will train a committee of CNNs for a single band. Since a machine with the latter parallel configuration is easier to find, boosting has an advantage over no-boosting but with a slight degradation in performance.

4.4. *Discussion*

While the Prague texture dataset provides a methodology of comparing different segmentation techniques, it is important to understand the performance on curved region boundaries. Therefore, the proposed method, with no retraining, is used to segment remote sensing images captured using ALI sensor, as presented in Figures 14(a)–,(f). Before segmenting ALI images, they are pre-processed by

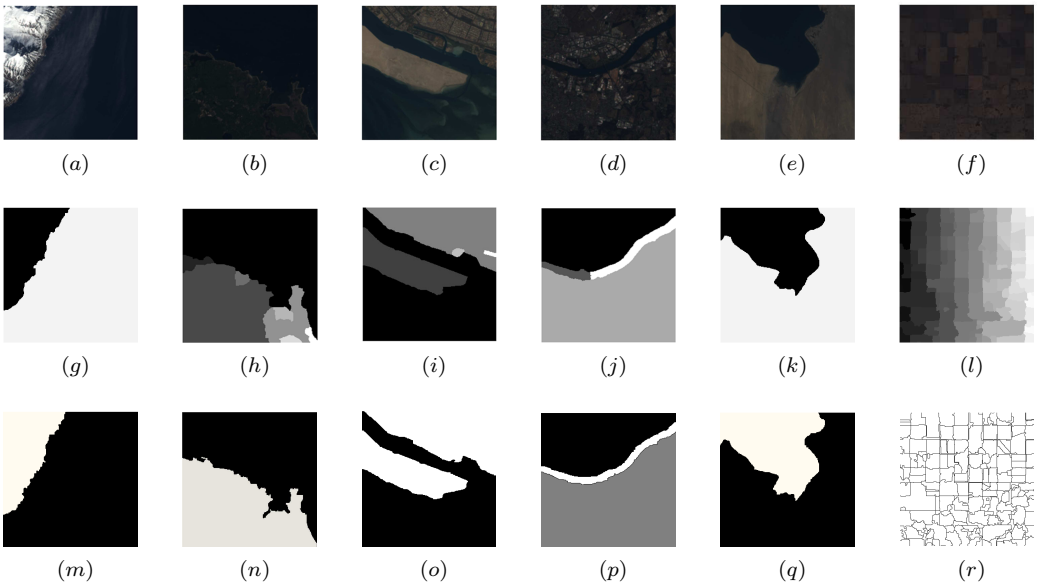


Figure 14. Segmentation results of the proposed method on remote sensing images captured using ALI sensor; (a)–,(f) input images, (g)–,(l) MSCNNs, and (m)–,(r) eCognition.

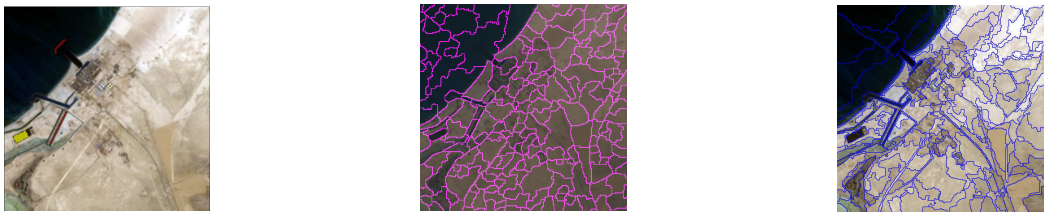


Figure 15. Segmentation in images with fine details of interest; (a) input image, (b) MSCNNs, and (c) eCognition.

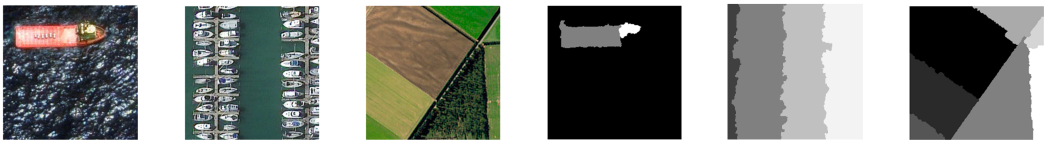


Figure 16. Segmentation results of the proposed method on remote sensing images captured using WorldView-2 sensor; (a)–,(c) input images and (d)–,(f) MSCNNs.

converting pixel values to exoatmospheric Top-Of-Atmosphere (TOA) reflectance as documented in (Chander, Markham, and Helder 2009). The conversion reduces scene-to-scene variations. The depth (scale) parameter, λ, and the merging threshold are set individually for each image in order to produce a subjectively appealing segmentation as reference segmentation do not exists. Results show that even though the proposed method is trained on straight boundary lines, it provides a good accuracy on curved region boundaries as depicted in Figures 14(g)–,(l). This means that the CNN can be successfully trained using a synthetic dataset (with curved boundaries if possible) and the time-consuming process of manual generation of training datasets can be avoided. In visual comparison to images segmented using eCognition method, presented in Figures 14(m)–,(r), segmentation results are comparable to the proposed method yet slightly better. However, the proposed

method produced a better segmentation in the last case (Figure 14(f)) where crop fields are uniformly segmented as rectangular regions. These results show that the proposed method managed to learn features and produce results similar to methods with an explicit set of features. Since regions in the Figure 14 are relatively large, another test is conducted in order to subjectively evaluate the effectiveness of the proposed method in segmenting small regions (demonstrated in Figure 15). In fact, Figure 15(b) demonstrates that the proposed method is capable of segmenting small regions (e.g., yellow-highlighted region in Figure 15(a)). It fails, however, to detect thin regions such as those highlighted in blue in Figure 15(a). On the other hand, eCongition is better in detecting such thin regions as demonstrated in Figure 15(c). In the second test, the proposed method, with no retraining, is used to segment images captured by WorldView-2 sensor. The aim here is to test the applicability of the proposed method on other sensors. Since WorldView-2 images have three bands only, the structure of the proposed method is reduced to contain bands of similar characteristics and the rest are ignored. A subjective evaluation of results in Figure 16 indicates that the proposed method has produced good segmentations despite being trained on ALI images. It is to be noted, though, that the best segmentation is expected when the proposed method is trained using WorldView-2 images.

The proposed method has several advantages over other techniques. The first advantage is its ability to present a hierarchical segmentation with no parameters to manually set. A segmentation in the hierarchy can be selected by fine-tuning two parameters in order to fit a particular application: the scale parameter, λ, and the merging threshold. The process of selecting these parameters is interactive yet fast as these parameters are associated with post-processing steps and does not require rerunning the algorithm (as in eCognition for example). One more advantage is that the proposed method produces a hierarchical segmentation which allows to segment large areas; forests, deserts, urban, water etc. or classify at a finer level such as marina, different crop fields, etc. without rerunning the segmentation technique. Also, unlike the work in (Ciresan et al. 2012), boundary pixels can be of zero pixels and do not need to have a particular appearance (such as in membrane pixels). Further, extracting patches at different scales and downscale them to a unified size improves results due to multi-scale analysis without the need to redesign the network or increase its computational complexity. Yet, it slightly increases the computational complexity of the pre-processing step.

The limitation in any supervised machine learning technique, as the case here, is the relatively long training time where it takes around 8.5 hours to train the whole structure for 100 epochs on the training set of around 150,000 samples in a parallel environment. However, the fast running time counterbalances it as it takes around 126 seconds to process a whole 512×512 image (a total of 262144 samples) in the same environment. It is to be noted, though, that the current implementation is a Matlab CPU implementation (Matlab 2014a running on 64-bit Windows 7 Professional on 2.40GHz CPU with 16 GB of RAM) which is not optimized for speed. It is expected that a GPU implementation would significantly reduce the time (Ciresan et al. 2012). Another limitation is the generation of a training dataset. Results, presented in this paper, suggest that a synthetic generation of training data could be sufficient and, hence, the inconvenient manual generation can be avoided. The third limitation is the selection of network parameters in terms of number and types of layers, number of filters at each layer, etc. This limitation is true for almost all supervised machine learning techniques such as Artificial Neural Network (ANN) and Support Vector Machine (SVM). The exhaustive search is a

widely used strategy in the field and CNN is no exception. However, once these parameters are set, they can be fixed for the particular application.

5. Conclusion

In this paper, a novel segmentation method for remote sensing images is presented. The proposed method exploits a committee of CNNs in order to detect region boundaries across scales. This is achieved through resizing input patches in a pyramid structure. Resizing the patches and not the original band serves two purposes. First, it assures one-to-one correspondence across all scales. Moreover, it allows all CNNs to share the same architecture thus eliminating the need to design different architectures for the different patch sizes. The improvement of adding multi-scale analysis is evident in the series of tests presented.

In order to overcome the limitations of the proposed method, a future direction is to implement the proposed method on parallel GPU implementation in order to reduce the training time. In addition, the incorporation of spectral and shape information within the framework can be used to improve the robustness of the proposed method.

Acknowledgement

The authors would like to acknowledge the support of His Highness Sheikh Mohamed bin Zayed Al Nahyan Program for Postgraduate Scholarships (Buhooth) under which the current research is conducted.

References

Abdel-Hamid, O., A Mohamed, H. Jiang, and G. Penn. 2012. "Applying Convolutional Neural Networks concepts to hybrid NN-HMM model for speech recognition." In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 4277–4280. Mar..

Baatz, M., and A. Schape. 2000. "Multiresolution segmentation: an optimization approach for high quality multi-scale image segmentation." *Angewandte Geographische Informationsverarbeitung XII* 12–23.

Basaeed, E., H. Bhaskar, and M. Al-Mualla. 2013. "A spectral water index based on visual bands." In *Image and Signal Processing for Remote Sensing XIX,* Vol. 8892889219. Dresden, Germany. Sep..

Chander, G., B. L. Markham, and D. L. Helder. 2009. "Summary of current radiometric calibration coefficients for Landsat MSS, TM, ETM+, and EO-1 ALI sensors." *Remote sensing of environment* 113 (5): 893–903.

Chen, X., S. Xiang, C. Liu, and C. Pan. 2014a. "Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks." *IEEE Geoscience and Remote Sensing Letters* 11 (10): 1797–1801.

Chen, Y., Z. Lin, X. Zhao, G. Wang, and Y. Gu. 2014b. "Deep Learning-Based Classification of Hyperspectral Data." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 7 (6): 2094–2107.

Ciresan, D., A. Giusti, L. Gambardella, and J. Schmidhuber. 2012. "Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images.." In *Neural Information Processing Systems (NIPS),* 2852–2860. Dec..

Ciresan, D., A. Giusti, L. M. Gambardella, and J. Schmidhuber. 2013. "Mitosis detection in breast cancer histology images with deep neural networks." In *Medical Image Computing*

and Computer-Assisted Intervention (MICCAI 2013), Vol. 8150 of *Lecture Notes in Computer Science* edited by K. Mori, I. Sakuma, Y. Sato, C. Barillot, and N. Navab. 411–418. Springer Berlin Heidelberg.

Ciresan, D., U. Meier, and J. Schmidhuber. 2012. "Multi-column deep neural networks for image classification." In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 3642–3649. IEEE.

Collobert, R., and J. Weston. 2008. "A unified architecture for natural language processing: Deep neural networks with multitask learning." In *Proceedings of the 25th international conference on Machine learning,* 160–167. Jul..

D'Elia, C., G. Poggi, and G. Scarpa. 2003. "A tree-structured Markov random field model for bayesian image segmentation." *IEEE Transactions on Image Processing* 12 (10): 1259–1273.

Gaetano, R., G. Scarpa, and G. Poggi. 2009. "Recursive Texture Fragmentation and Reconstruction segmentation algorithm applied to VHR images." In *2009 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2009),* Vol. 4IV–101–IV–104. Jul..

Guigues, L., J. Cocquerez, and H. Men. 2006. "Scale-Sets Image Analysis." *International Journal of Computer Vision* 68 (3): 289–317.

Hu, Z., Z. Wu, Q. Zhang, Q. Fan, and J. Xu. 2013. "A Spatially-Constrained Color-Texture Model for Hierarchical VHR Image Segmentation." *IEEE Geoscience and Remote Sensing Letters* 10 (1): 120–124.

Jensen, J. 2004. *Introductory Digital Image Processing.* 3rd ed. Prentice Hall.

Jung, Chanho, and Changick Kim. 2010. "Segmenting Clustered Nuclei Using H-minima Transform-Based Marker Extraction and Contour Parameterization." *IEEE Transactions on Biomedical Engineering* 57 (10): 2600–2604.

Kim, S., W. Lee, D. Kwak, G. Biging, P. Gong, J. Lee, and H. Cho. 2011. "Forest Cover Classification by Optimal Segmentation of High Resolution Satellite Imagery." *Sensors* 11 (12): 1943–1958.

Krizhevsky, A., I. Sutskever, and G. E. Hinton. 2012. "Imagenet classification with deep convolutional neural networks." In *Advances in neural information processing systems,* 1097–1105. Curran Associates, Inc.

Lecun, Y., L. Bottou, Y. Bengio, and P. Haffner. 1998. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86 (11): 2278–2324.

Li, H. T., H. Y. Gu, Y. S. Han, and J. H. Yang. 2008. "An efficient multi-scale segmentation for high-resolution remote sensing imagery based on Statistical region merging and minimum heterogeneity rule." In *International Workshop on Earth Observation and Remote Sensing Applications, 2008 (EORSA 2008),* Beijing, China. Jul..

Li, L., J. Ma, and Q. Wen. 2007. "Parallel fine spatial resolution satellite sensor image segmentation based on an improved PulseCoupled Neural Network." *International Journal of Remote Sensing* 28 (18): 4191–4198.

Long, J., E. Shelhamer, and T. Darrell. 2015. "Fully convolutional networks for semantic segmentation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 3431–3440.

Marpu, P., M. Neubert, H. Herold, and I. Niemeyer. 2010. "Enhanced evaluation of image segmentation results." *Journal of Spatial Science* 55 (1): 55–68.

Mikes, S., M. Haindl, and G. Scarpa. 2012. "Remote sensing segmentation benchmark." In *2012 IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS),* 1–4. Nov..

Mnih, Volodymyr. 2013. "Machine learning for aerial image labeling." Ph.D. thesis. University of Toronto.

Myint, S., P. Gober, A. Brazel, S. Grossman-Clarke, and Q. Weng. 2011. "Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery." *Remote Sensing of Environment* 115 (5): 1145–1161.

Novack, T., T. Esch, H. Kux, and U. Stilla. 2011. "Machine Learning Comparison between WorldView-2 and QuickBird-2-Simulated Imagery Regarding Object-Based Urban Land Cover Classification." *Remote Sensing* 3 (12): 2263–2282.

Penatti, O., K. Nogueira, and J. dos Santos. 2015. "Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains?." In *The IEEE Confer-*

ence on Computer Vision and Pattern Recognition (CVPR) Workshops, 44–51. Boston, MA: IEEE. Jun.. http://homepages.dcc.ufmg.br/ jefersson/pdf/penatti2015cvprw.pdf.

Salembier, P., and L. Garrido. 2000. "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval." *IEEE Transactions on Image Processing* 9 (4): 561–576.

Scarpa, G., G. Masi, R. Gaetano, L. Verdoliva, and G. Poggi. 2013. "Dynamic Hierarchical Segmentation of Remote Sensing Images." In *Image Analysis and Processing ICIAP 2013,* edited by A. Petrosino. no. 8156 In Lecture Notes in Computer Science. 371–380. Springer Berlin Heidelberg.

Scarpa, G., G. Masi, L. Verdoliva, G. Poggi, and R. Gaetano. 2012. "Recursive-TFR Algorithm for Segmentation of Remotely Sensed Images." In *2012 Eighth International Conference on Signal Image Technology and Internet Based Systems (SITIS),* 174–181. Nov..

Scherer, D., A. Mller, and S. Behnke. 2010. "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition." In *Artificial Neural Networks ICANN 2010,* edited by K. Diamantaras, W. Duch, and L. Iliadis. no. 6354 In Lecture Notes in Computer Science. 92–101. Springer Berlin Heidelberg.

Schulz, H., and S. Behnke. 2012. "Learning object-class segmentation with convolutional neural networks." In *11th European Symposium on Artificial Neural Networks (ESANN),* Vol. 31. Bruges. Apr..

Sermanet, P., and Y. LeCun. 2011. "Traffic sign recognition with multi-scale Convolutional Networks." In *The 2011 International Joint Conference on Neural Networks (IJCNN),* 2809–2813. Jul..

Simard, P., D. Steinkraus, and J. Platt. 2003. "Best practices for convolutional neural networks applied to visual document analysis." In *2013 12th International Conference on Document Analysis and Recognition,* Vol. 2958–958. IEEE Computer Society.

Tarabalka, Y., J.C. Tilton, J.A. Benediktsson, and J. Chanussot. 2012. "A Marker-Based Approach for the Automated Selection of a Single Segmentation From a Hierarchical Set of Image Segmentations." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 5 (1): 262–272.

Tilton, J.C., Y. Tarabalka, P.M. Montesano, and E. Gofman. 2012. "Best merge region-growing segmentation with integrated nonadjacent region object aggregation." *IEEE Transactions on Geoscience and Remote Sensing* 50 (11): 4454–4467.

Trias-Sanz, R., G. Stamon, and J. Louchet. 2008. "Using colour, texture, and hierarchical segmentation for high-resolution remote sensing." *ISPRS Journal of Photogrammetry and Remote Sensing* 63 (2): 156–168.

Ungar, S.G., J.S. Pearlman, J.A. Mendenhall, and D. Reuter. 2003. "Overview of the Earth Observing One (EO-1) mission." *IEEE Transactions on Geoscience and Remote Sensing* 41 (6): 1149–1159.

Visa, A, K. Valkealahti, and O. Simula. 1991. "Cloud detection based on texture segmentation by neural network methods." In *IEEE International Joint Conference on Neural Networks,* Vol. 21001–1006. Nov..

Weng, Qihao, and Xuefei Hu. 2008. "Medium Spatial Resolution Satellite Imagery for Estimating and Mapping Urban Impervious Surfaces Using LSMA and ANN." *IEEE Transactions on Geoscience and Remote Sensing* 46 (8): 2397–2406.

Wu, Haibing, and Xiaodong Gu. 2015. "Towards dropout training for convolutional neural networks." *Neural Networks* 71: 1–10. http://www.sciencedirect.com/science/article/pii/S0893608015001446.

Xiaoying, J. 2009. "Segmentation-based image processing system." .

Yue, Jun, Wenzhi Zhao, Shanjun Mao, and Hui Liu. 2015. "Spectralspatial classification of hyperspectral images using deep convolutional neural networks." *Remote Sensing Letters* 6 (6): 468–477. http://dx.doi.org/10.1080/2150704X.2015.1047045.

Zhang, W., R. Li, H. Deng, L. Wang, W. Lin, S. Ji, and D. Shen. 2015. "Deep convolutional neural networks for multi-modality isointense infant brain image segmentation." *NeuroImage* 108: 214–224.

Zhao, W., Z. Guo, J. Yue, X. Zhang, and L. Luo. 2015. "On combining multi-

scale deep learning features for the classification of hyperspectral remote sensing imagery." *International Journal of Remote Sensing* 36 (13): 3368–3379. http://www.tandfonline.com/doi/full/10.1080/2150704X.2015.1062157.